REMARKS

Claims 1-32 currently remain in the application. Claims 33-97 have been cancelled. Claim 1 has been amended.

Rejections under 35 U.S.C. § 102

The Examiner rejected claim 1-11 under 35 U.S.C. § 102(e) as being unpatentable over Jackson, U.S. 2003/0069074. The rejection is respectively traversed.

Jackson describes a gaming machine and an associated software architecture. The gaming software architecture employs game software modules that are described in FIG. 3 as shared objects (see paragraphs 146,147 and 160-162, Figs. 2 and 3). As an example, in paragraph 147, it states,

> *The system handler 202 application of some embodiments may manage the gaming program shared objects by loading a single object at a time and executing the object. When another object needs to be loaded and executed, the current object may remain loaded or can be unloaded and the new object loaded in its place before the new object is executed. The various shared objects can pass data between objects by storing the data in nonvolatile storage 204. <u>For example, a "game.so" file may be a gaming program object file that is loaded and executed to provide operation of a feature set of a computerized wagering game, while a "bonus.so" gaming program object file is loaded and executed to provide a feature set of the bonus segment of play. Upon changing from normal game operation to bonus, the bonus.so is loaded and executed upon loading.</u> Because the relevant data used by each gaming program object file in this example is stored in nonvolatile storage 204, the data may be accessed as needed by whatever gaming program object is currently loaded and executing.*

In this paragraph, as well as paragraphs 146 and 160-162, the details as to where the logic resides that determines when to trigger a bonus game or where the logic resides that specifies the details necessary for generating a presentation for the game and/or the presentation for the bonus game (e.g., displaying particular images and generating particular sounds in the case of a video gaming machine using the device drivers/handlers 210 in FIG. 3) is not provided.

In the present invention, as recited in claim 1 for instance, a game manager is used to control an invocation of a plurality of stages between a start of the game of chance and an end of the game of chance. The game manager determines when to invoke each stage in the plurality of stages. The logic for the game manager is separate from the logic for the game stage and the one or more additional stages. Further, the game manager is operable to invoke a variable number of

7

stages in each play of the game of chance. For instance, the total number of possible stages in a game can be varied.

One advantage of employing the game manager utilizing logic separate from the game stage and separate from the one or more additional stages is that the logic for when to invoke different stages or additional stages can be added without modifying the logic that describes the game flow in the game stage. For example, in FIGs. 7A and 7B, a bonus stage is triggered after a game stage. In the present invention, the logic for determining when to trigger a bonus stage or how many bonus stages to trigger in the play of game of chance may be modified without modifying the game stage. Thus, the format of the game can be varied without having to submit the game stage code for approval by a gaming jurisdiction. It is not clear in Jackson whether the gaming machines endowed with this capability.
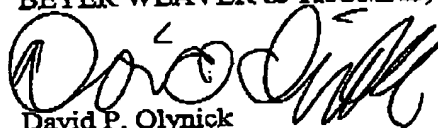
In addition, in each stage, one or more game states is generated and for each game state a corresponding presentation state is generated where each presentation state specifies operations of output devices on the gaming machine. The logic for the game states and the presentation states in each stage are configured such that modifying the logic for each presentation state does not affect the logic of its corresponding game state. For instance in one aspect of the present invention, a game state may be embodied as a "game state shared object" and a corresponding presentation state may be embodied as a "presentation state shared object."

One advantage of this architecture is that the presentation state logic can be modified or replaced without affecting its associated game state. Further, the theme or feel of a game can be modified by substituting different presentation state objects for corresponding game states without changing the underlying game state logic. It is not clear in Jackson whether the gaming machine endowed with this capability.

Therefore, for at least these reasons applicant believes Jackson can't be said to anticipate the remaining claims and the rejection is believed overcome thereby.

Applicant believes that all pending claims are allowable and respectfully requests a Notice of Allowance for this application from the Examiner. Should the Examiner believe that a telephone conference would expedite the prosecution of this application, the undersigned can be reached at the telephone number set out below.

Respectfully submitted,
BEYER WEAVER & THOMAS, LLP

David P. Olynick
Reg. No.: 48,615

P.O. Box 70250
Oakland, CA 94612-0250
(510) 663-1100, ext. 231

8